STAT

DATE: May 24, 1982

NOTE TO:

SUBJECT: Advantages of IBM over Burroughs

Ed:

I have prepared two lists. The first list mentions specific advantages associated with an IBM approach. The second list is disadvantages associated with Burroughs. In addition, for most cases, adopting Burroughs would produce a disadvantage in failing to capitalize on the IBM advantages. The lists are in no particular order. I have put an asterisk by items which are mostly relevant to CIA.

IBM Advantages

1. Use of Existing Software.

2. Ability to apply related developments to SAFE.

3. *Ability to apply SAFE-developed software to non-SAFE environments.

4. *Similarity to DDO's ALLSTAR Upgrade.

5. Rich community of third-party vendors and other users developing software oriented to information processing.

6. Availability of alternative sources for hardware (PCM's and peripherals). Note: In some cases, such as STC paging devices and Comten front-ends, the plug-compatible devices might exceed the capabilities of the IBM devices.

7. *Reassignment of ADPE ("CPU Shuffle").

8. *Ability to share backup.

9. *Interoperability with existing base of equipment (e.g. file sharing, device sharing, computer-to-computer links etc.)

10. *Opportunities for load leveling.

11. *Existing IBM-oriented staff.

12. Availability of IBM-oriented consultants and job seekers.

13. *In-place IBM-oriented development facilities.

14. Ability to run PMO and other interim SAFE packages on selected equipment.

15. *Ability to share risks with other projects (MHF, ALLSTAR, VM).

16. Technological superiority of IBM and Amdahl.

<u>Burroughs Disadvantages</u>

1. Equipment size.

2. Power consumption

3. Poor development tools.

4. Inadequate multi-tasking environment (limited to about 50 tasks).

5. Memory limitations on current hardware (6 MB).

6. Poor disk technology.

7. Poor support for shared disks.

8. Banking orientation (versus Information Processing).

DATE: May 24, 1982

STAT

NOTE TO: [   ]

SUBJECT: Comments on SAFE STAB Paper

Ed:

I have read the STAB paper and these are my comments. On the whole, I would say that the paper was disappointing. While there are scattered points that I agree with, the paper fails to focus on the really important issues which would facilitate decision making. I presume that the intent of the group, and of the paper, is to provide Harry with assistance in interpreting the various inputs from the Audit Team, CSPO, RSO, ODP, STAP, and TRW. In a real sense, all he got was more input, and not particularly relevant or helpful input, at that.

What the group appears to have done is to put on the table yet another architecture. Granted, it might be very attractive to have a system built in such a manner as to be able to draw upon relevant developments from a number of vendors. Indeed, the academic community (and to some degree, NSA) has successfully integrated different computing environments in networks which support moderately large numbers of terminals. Despite a certain surface appeal, the suggested approach actually represents substantially greater risk than either a pure Burroughs- or IBM-oriented approach. The ARPA and other comparable networks were not designed to provide a single system, but rather a loose federation of cooperating systems. I know of no comparable development to that of SAFE having been successfully implemented in such a manner. The closest government system to the STAB suggestion (similar in intent but certainly not in scale) is the DIA's NMIC system, which is hardly a model of an efficiently executed system development.

I believe the STAB, as TRW before them, has made a fundamental error in presuming that systems built for the government should be based upon radical departures from vendor-supported machine architectures. Granted, we have found ways of extending the IBM architecture, but that has typically been accomplished through slow evolution based upon supported concepts. We should not be in the buiness of developing "contemporary innovative architectures", but should be prepared to exploit such developments when they are developed and supported by the vendors.

The underlying theme in the paper is that we could couple a bunch of UNIX-based (presumably DEC) machines together along with other "appropriate" hardware for other functions. What they have done is recommend yet another "family" of hardware and software -- instead of IBM and Burroughs, we should use DEC's (or rather RAND and NSA's) equipment. This proposal is risky for several reasons.

First of all, computer networking, except in the universities, is in its infancy. So-called "standard interface arrangements" are not all that standard. Hetrogeneous networks, such as the ARPANET and X.25, are not supported adequately by the major vendors to insure that software developed for the individual machines can be adequately operated through

the network. For example, there is neither an ARPANET or X.25
connection for VM/370. We could get a Burroughs ARPANET interface
(developed for the Platform network), but that would be NSA-supported
software. Even DEC really supports DECNET, although software could be
obtained from the ARPA community for attaching DEC minis to the ARPANET.

The UNIX system was seriously considered by the Agency a few years ago
as a means of supporting distributed processing. That approach was
quickly abandonned, however, when it became apparent that there were
significant support problems associated with that system. We found
total disarray in the UNIX community, when contrasted with standard
vendor-supported software. At the time we looked, there were at least
four different versions of the system, including an entirely
restructured UNIX to support the Programmer Work Bench (PWB), and no
real sense of direction, due to the widely different community of
users. That situation has changed somewhat, because of certain
developments at Western Electric, DEC, and Interactive System
Corporation, but is still far from ideal.

UNIX offers some attractive capability for editing, composing, job
submission, and program development. It has a very fundamental
attractiveness, machine independence, as well as a very flexible command
interface and nice multi-tasking support. I, for one, don't think it is
all its cracked up to be, and I believe that there are several systems
which offer comparable or greater facilities. Most of the best
user-oriented characteristics are in VM; certainly the MULTICS system is
considerably better than UNIX (which was spawned from it), and I would
definitely prefer DEC's VAX/VMS to UNIX.

I concur with the notion that the SAFE system is not innovative from a
contemporary perspective. I think that the offerings of the Xerox
Corporation are far more illustrative of that fact than are the systems
operated at Rand and NSA.

The STAB report failed to account for the agency context into which the
SAFE system must fit. The reason that the DEC hardware works so well in
the NSA and Rand environments is that they have long traditions of operating
DEC equipment, and their computer people have evolved with the DEC
evolution. We, at least at CIA, have that same relationship to a
vendor, IBM. Incidentally, we have achieved very much the same type of
interconnection between our IBM mainframes as RAND and NSA have with
their equipment.

A STAB suggestion that the system might be built more on the concept of
"localized files and inter-terminal communication" demonstrates a lack
of understanding of the fundamental nature of the SAFE requirements --
sharing of a very large quantity of centralized data -- and
fails to come to grips with the security and operational difficulties
which would result from a widely distributed system.

Loosely interpreted, I concur with the STAB recommendation to exploit
existing software -- fortunately a good bit of what we are looking for exists
somewhere on IBM. I sensed a profound lack of understanding of the
notion of "IBM compatible" from the stand-point of plug-compatible

mainframes and peripherals. While not, by any means, the best systems software available, IBM software is clearly adequate for the variety of functions envisioned for SAFE. I even see how it might be possible, if it were truly desired, to use specialized machines, such as DEC, in an IBM-oriented network.

The STAB dealt somewhat unclearly with the Safe User Language by saying that "it may be impossible to impement a comprehensive user langaguge without segmenting it". The SAFE Audit Team went further, suggesting that the need to develop a single, integrated language, had driven the whole project, and that the goal of a comprehensive language ought to be dropped.

In the paragraph on test beds, they fail to acknowlege the incremental approach suggested by the Audit Team, and appear to be commenting on the original proposal of SAFE, instead. The final suggestion that we develop an overall system architecture which would permit a phased approach is actually quite close to the recommendation of the Audit Team. The comment that the system is "understood as hardware arrangements" is, again, more appropriate to the original SAFE untertaking, rather than the recent suggestions for improving it.

Finally, the observation that C and D systems developed in parallel would "lead inevitably to divergence" is worth noting. I don't believe that anyone has suggested that the two systems must be identical. To the contrary, the Audit Team found that the DIA was being constrained by the two systems' being developed together. I think that the degree of divergence will be a reflection of the degree of dis-similarity in the requirements, and will not be at all unhealthy. The ability to run CIA-developed software at DIA and vice-versa can certainly be maintained, just as several thousand different installations share individual developments in their IBM environments today.

STAT